Improving IEEE 802.15.4 for Low-latency Energy-efficient Industrial Applications

Feng Chen

Computer Networks and Communication Systems
University of Erlangen-Nuremberg, 91058 Erlangen
feng.chen@informatik.uni-erlangen.de

Abstract The IEEE 802.15.4 standard for LR-WPANs is becoming a de-facto standard for Wireless Sensor Networks (WSNs) applications in industrial fields. In this paper, we evaluate the latency performance of the IEEE 802.15.4 protocol based on a typical industrial scenario: a star network with 20 devices that send short messages (1 Byte) to the PAN coordinator. We analyzed the behavior of the GTS mechanism in the standard analytically. The results reveal essential limitations of the standard for low-latency applications in automation environments. According to our findings, we propose an enhanced protocol version that fully supports industry demands on low-latency communication. Our protocol version uses the original physical layer and, thus, can be implemented conveniently using cheap IEEE 802.15.4 hardware. The analytical results prove that we are able to meet the guaranteed low latency of 10 ms as specified by typical automation environments.

1 Introduction

Wireless Personal Area Network (WPAN) technology, which supports short-range, low-cost and energy-efficiency networking, is widely used as a base for Wireless Sensor Networks (WSNs). Their ease of deployment and the widespread use makes WSNs also attractive for a number of commercial, especially industrial applications. For example, the department "Automation and Drives, A&D" of Siemens AG is currently evaluating wireless technology in the field of industrial automation. In the domain of WSNs, a number of Medium Access Control (MAC) protocols, for example, Sensor MAC (S-MAC) [7], have been proposed. However, these MAC protocols have not yet made their way into commercial applications. In contrast, the IEEE 802.15.4 standard [2] has been developed and is accepted by industrial users. It provides specifications for the Physical Layer (PHY) and Medium Access Control (MAC) sublayer for the use in LR-WPANs. Products that implement this standard are commercially available at an acceptable low cost.

In this paper, we study the applicability of LR-WPAN techniques in industrial control applications. As energy consumption is not the most critical parameter, our objective here is to evaluate the latency performance of the IEEE 802.15.4 protocol relevant to the intented real-time application scenario.

We show that the protocol specification does not fulfill industry demands for low-latency transmission. Therefore, we propose modifications of the standard to circumvent these limitations.

The contributions of this paper can be summarized as follows. Based on a specific application scenario as required in industrial applications as depicted in Section 1.1, we deeply analyze the IEEE 802.15.4 protocol with primary focus on low-latency data transmission. This evaluation is performed using analytical methods (see Section 3). We identify the protocol-inherent limitations that prevent its use in delay sensitive industrial applications and propose selected protocol modifications that allow to still use off-the-shelf hardware. The modifications including an appropriate analysis are depicted in Section 4.

1.1 An Industrial Case Study

The typical application scenario for automation environments to be studied in this paper is described as follows. The numbers in brackets are examples from typical automation projects of Siemens AG. A number of sensor nodes ($n = 20$) are scattered within an area and associated to a central node to form a star network, which is continuously monitoring industrial processes. Once a certain device detects that particular sensor readings exceed a predefined threshold, a short alarm message (1 Byte) must be sent by the device to the central node within a given time frame (guaranteed low latency $d_{GUA} < 10$ ms). Such a time limit is a hard real-time requirement, thus, the network needs to be able to handle also the worst case, when all devices generate alarm signals at the exactly same time. In addition, to prolong the lifetime of the monitoring sensor network, all the devices need to enter a sleeping mode if no critical events are detected.

1.2 Related Work

Recently, a number of papers have been published on performance analysis of IEEE 802.15.4 networks [6,8]. Nevertheless, most of this work focuses on typical WSN applications rather than industrial automation domains.

In [3], Kim et al. proposed priority-based scheme comprising *Frame Tailoring* (FRT) and *Priority Toning* (PRT) to reduce latency in event-monitoring IEEE 802.15.4 networks. However, these methods are contention-based, which still cannot provide guaranteed latency bounds. In [4], the delay bounds guaranteed by the IEEE 802.15.4 GTS allocations have been analyzed for real-time WSNs using the analytical Network Calculus formalism. Based on this analysis, the authors pointed out the limitations of the explicit GTS allocation in IEEE 802.15.4 and proposed an *implicit GTS Allocation Mechanism* (i-GAME) [5]. However, the new approach improves the bandwidth utilization of the original GTS mechanism at the cost of increasing guaranteed delay bounds, which is not applicable for industrial applications with very strict real-time requirements.

2 Overview of IEEE 802.15.4

In this section, a brief overview to the IEEE 802.15.4 protocol is provided. For a more detailed description of the protocol, the reader is recommended to refer to the protocol standard documents [2] and to [1].

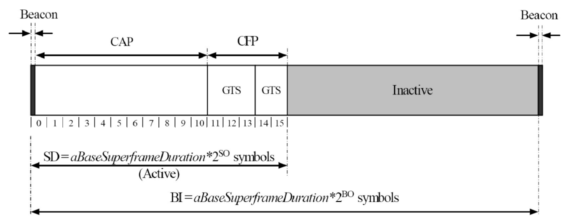

Figure 1. IEEE 802.15.4 superframe structure

The IEEE 802.15.4 WPAN standard supports two network topologies, a star and a peer-to-peer topology. In this work, we consider only star networks, in which the communication occurs only between end devices and the PAN coordinator. In order to synchronize the communication at MAC layer, the IEEE 802.15.4 PAN can optionally operate in the so called *beacon-enabled* mode. In this case, a superframe structure is used as shown in Fig. 1. Each *superframe* is bounded by periodically transmitted beacon frames and consists of two parts, an active portion and an inactive portion. In order to save energy, nodes may enter a low-power (sleep) mode during the inactive portion. The superframe structure is specified by the values of two MAC attributes: the *macBeaconOrder* (BO) and the *macSuperframeOrder* (SO), both of which determine the length of the beacon interval (BI) and the length of the active portion of the superframe (SD), respectively. The relation of BO to BI and the relation of SO to SD are shown in Fig. 1. The *aBaseSuperframeDuration* equals to 960 symbols. PANs that wish to use this superframe structure (referred to as a beacon-enabled PANs), shall set BO to a value between 0 and 14 and SO to a value between 0 and the value of BO, resulting in the range of BI and SD between 15.36 ms and 251.7 s at the 2.4 GHz band. If BO=15, PANs operate in a so-called *nonbeacon-enabled* mode without using the superframe structure.

The active portion of the superframe shall be divided into 16 equally spaced *superframe slots*. The duration of one superframe slot is calculated by $2^{SO} \times$ *aBaseSlotDuration*, where the default value of *aBaseSlotDuration* is 60 symbols. There are three parts in the active portion: a beacon, a Contention Access Period (CAP), and a Contention-Free Period (CFP). In the CAP, all data transmissions shall follow a successful execution of a slotted CSMA-CA algorithm.

For low-latency applications, the PAN coordinator may dedicate portions of the active superframe to that application, which are called guaranteed time slots (GTSs). They allow the channel access in TDMA-like fashion. The GTSs are located between the CAP and the inactive portion to form the Contention-Free

Period, as shown in Fig. 1. The PAN coordinator may allocate a maximum of seven GTSs at the same time, and one GTS may occupy more than one superframe slot. The CFP containing all allocated GTSs shall grow or shrink dynamically within the active portion. However, a minimum length of the CAP with *aMinCAPLength* (440) symbols must be guaranteed and remains for contention-based access of other networked devices or new devices wishing to join the network.

3 Analytical Worst Case Estimation

The IEEE 802.15.4 standard provides both contention-based and contention-free (GTS) channel access methods, while the first one cannot provide any guaranteed quality of service. Therefore, we study only the GTS scheme in this paper. In this section, the behavior of the GTS mechanism is first evaluated according to the protocol standard of a maximum of seven GTSs. The calculation results reveal several limitations in the original GTS mechanism and motivated us to remove those constraints. The consequent recalculations show the need for further improvements in the standard.

3.1 Standard Protocol Behavior

As shown in Fig. 1, a beacon interval (l_{BI}) consists of the following fields: a beacon (l_B), a SIFS (l_{SIFS}), the CAP (l_{CAP}), up to seven GTSs ($n \times l_{GTS}$), and the inactive portion (l_{SLP}). Each GTS is composed of an integer number of superframe slots ($n \times l_{SS}$) and should accommodate at least one complete transaction (l_{TR}), including one data transmission (l_D) and a SIFS (l_{SIFS}). Thus, the length of a beacon interval can be calculated as follows:

$$l_{BI} = l_B + l_{SIFS} + l_{CAP} + n \times l_{GTS} + l_{SLP} \tag{1}$$

For a certain scheme of GTS allocation, the guaranteed latency, which is measured by the maximum latency among all the GTS transmissions under all traffic conditions, can be estimated through analyzing the worst case. The worst case would happen in the network if a message is generated at a device during its own GTS slot. At this time, the device cannot transmit the message immediately and must buffer the message. The buffered message must wait for one beacon interval until the start of the corresponding GTS in the next superframe and needs a transaction period to get transmitted. Therefore, the guaranteed latency denoted as l_G under the worst case is bounded by the sum of one beacon interval and one transaction period, which is formulated as follows:

$$l_G = l_{BI} + l_{TR} \tag{2}$$

In the following, we consider a maximum of seven GTS allocations in a star network with seven devices and a PAN coordinator and calculate the minimum guaranteed latency for transmitting alarm messages with exactly one byte payload. Addressing information is not needed because only a specific device that own this GTS is allowed to send at this time. Since the energy consumption is not the main interest in this calculation, the BO is set equal to the SO to eliminate the inactive portion. According to (2), the beacon interval should be set as small as possible to achieve lower latency. On the other hand, the active portion, which is determined by SO, must be set long enough to accommodate seven GTSs in the CFP and maintain a minimum CAP length of 440 symbols, denoted as l_{minCAP}, according to the standard. Based on these rules, some duration values calculated according to the standard and listed in Table 1 are used to choose the minimum (BO,SO) combination.

Table 1. Duration Parameters

Symbol	Description	Value
l_B	length of beacon transmission	34 symbols
l_D	length of data transmission	40 symbols
l_{SIFS}	short interframe space	12 symbols
l_{TR}	length of one transaction	52 symbols

If both, BO and SO, are set to 0, l_{BI} is equal to 960 symbols. The resulting l_{SS} of 60 symbols is bigger than l_{TR}. Therefore, one GTS l_{GTS} is allocated with one superframe slot and equals to 60 symbols. According to (1) and the rule of minimum CAP, the minimum required beacon interval (l_{minBI}) can be calculated as follows:

$$l_{minBI} = l_B + l_{SIFS} + l_{minCAP} + 7 \times l_{GTS} = 906 \text{ symbols}$$

This calculated minimum length is smaller than the actual beacon interval of 960 symbols, which is obtained using the formula in Fig. 1 with BO equal to 0. Therefore, the setting of (BO,SO) to (0,0) can support seven GTS allocations. The guaranteed latency can be calculated according to (2) using l_{BI} with 960 symbols:

$$l_G = l_{BI} + l_{TR} = 1012 \text{ symbols} \Rightarrow d_G = l_G/(62.5 \text{ ksymbols/s}) = 16.2 \text{ ms}$$

16.2 ms is the smallest one that the standard protocol can achieve among all the settings. However, this result does not satisfy our requirement of 10 ms, when only seven devices are considered in the network.

3.2 Limitation Analysis

By evaluating the behavior of the standard protocol, we identified the following limitations for the GTS mechanism in IEEE 802.15.4:

Firstly, the constraint of *maximum seven GTSs* limits the number of devices involved in the GTS usage. A relatively short active period can even reduce this number. Once the capacity of GTS allocations is full, other devices desiring for GTS slots have to wait until some of the previously allocated GTSs have been released. The allocation and deallocation process will consume a considerable time, which would be intolerable for real-time applications. Secondly, the *minimum CAP length* (440 symbols) defined by the standard further restricts the available length of the CFP for GTS allocation and introduces an extra latency to GTS transmissions. Finally, one GTS can only consist of an integer number of superframe slots. The length of one superframe slot calculated by $2^{SO} \times aBaseSlotDuration$ grows exponentially with an increasing SO. This may lead to an inefficient bandwidth use, when the required bandwidth is much smaller than that the minimum GTS provides.

3.3 Removal of Limitations

In our first try, we ignore the restriction to seven GTSs per beacon interval. Additionally, the required minimum CAP and the optional inactive portion are removed, i.e. $l_{CAP} = l_{SLP} = 0$. Each GTS l_{GTS} is allocated with the standard-defined minimum length of 60 symbols, which has been set bigger than l_{TR} with 52 symbols to guarantee one complete transaction in the GTS. In this slightly improved protocol version, BI and SD are not determined by (BO,SO) combinations anymore. All other mechanisms including beaconing and the frame structure are kept.

Now we re-evaluate the latency performance of the improved GTS mechanism allowing the required number of 20 GTSs. The resulting beacon interval can be calculated as follows using (1), in which the l_{CAP} and the l_{SLP} are eliminated and n is set to 20.

$$l_{BI} = l_B + l_{SIFS} + 20 \times l_{GTS} = 1246 \text{ symbols}$$

The guaranteed latency can be calculated as follows:

$$l_G = l_{BI} + l_{TR} = 1298 \text{ symbols} \Rightarrow d_G = 20.77 \text{ ms}$$

which is bigger than 10 ms. Therefore, the further improvements are required. Since 20 GTSs with a total of 1200 symbols have contributed the majority of the beacon interval, we need to further reduce the length of each GTS. As described previously, each GTS has a limitation on the minimum allocation unit with a superframe slot. If we remove this constraint, each GTS can be allocated with an exact bandwidth for one complete transaction by setting l_{GTS} equal to l_{TR}. Thus, the guaranteed latency can be calculated as follows:

$$l_G = l_B + l_{SIFS} + n \times l_{TR} + l_{TR} \tag{3}$$

Thus, the maximum latency for $n = 20$ nodes is:

$$l_G = l_B + l_{SIFS} + 21 \times l_{TR} = 1138 \text{ symbols} \Rightarrow d_G = 18.21 \text{ ms}$$

The calculated latency is still too large. If looking at the value of l_D listed in Table 1, we can find that transmitting an alarm message with only one byte payload needs an overhead of 38 symbols added by the MAC and the PHY. This big overhead consumes most of the bandwidth and needs to be reduced.

4 Low-Latency Protocol

In this section, we present an IEEE 802.15.4 based protocol version that has been improved explicitly for the industrial real-time application described in Section 1.1. To achieve better hardware comparability, the PHY layer is completely preserved. Our improvements on the IEEE 802.15.4 MAC sublayer mainly include two aspects, the modification of the superframe structure and the reduction of the MAC overhead. In the following, we introduce our protocol in detail.

4.1 TDMA-based Superframe Structure

Each superframe consists of an IEEE 802.15.4 compliant beacon, n GTSs and $n+1$ interframe spaces. The frame structure is shown in Fig. 2. We completely removed the contention access period, therefore, instead of allocation in a request-reply fashion as defined in the standard, all GTSs need to be preallocated to each of the n devices. In our application, we assume that only unacknowledged uplink transmissions from devices to the PAN coordinator will occur in the GTS. Thus, we distinguish two interframe space types. An IFS with length (l_{IFS}) equal to *aTurnaroundTime* symbols, which is equal to 12 symbols in the standard, is used before and after the beacon frame to guarantee that radios of the PAN coordinator and devices can switch between RX and TX state. For the interframe space between neighboring GTSs, a SIFS with a shorter length (l_{SIFS}) of 4 symbols is defined. We assume that this value is long enough for two consecutive transmissions, because the PAN coordinator always stays in a receiving state during this period. In enhanced protocol version, the interframe space has been separated from the GTS, which differs from the way described previously, and each GTS l_{GTS} will be allocated with only the length of one data transmission l_D.

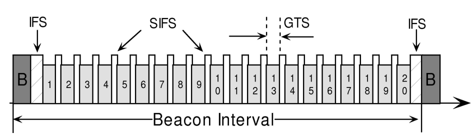

Figure 2. TDMA-based superframe structure

In the used star topology, the communication is initiated by the PAN coordinator through broadcasting a beacon frame, which carries the information of the deployed superframe structure including the beacon interval and the position of the GTS preallocated to each device. Upon reception of the first beacon, each device can be configured to have one of the following two options:

Beacon tracking enabled – The device keeps in sync with the PAN coordinator through tracking the beacons transmitted by the PAN coordinator. For this purpose, the device has to wake up a short period of time before the scheduled arrival of each beacon. Upon reception of the beacon, it goes back to sleep and wakes up again only in its own GTS if it has a message to send.

Beacon tracking disabled – To save as much energy as possible, the device can go to sleep immediately after receiving the first beacon. It will wake up again only when a new message is generated. To transmit the message, the device needs first to resynchronize to the PAN coordinator by tracking the next coming beacon. Upon reception of one beacon, the node can locate its own GTS and send the message within this GTS. Afterwards, the node returns to sleep again.

In all cases, the PAN coordinator has to stay awake all the time to transmit beacons and to receive data from the devices. In industrial applications, such a PAN coordinator is assumed to be powered sufficiently. Therefore, no energy-efficiency is considered for the PAN coordinator in our protocol.

4.2 Data Frame Format without MAC Header

As discussed in the previous section, to transmit one byte payload, the standard protocol adds a relatively huge overhead of 38 symbols at the MAC and the PHY. Therefore, another goal in designing our protocol is to reduce such big overheads. Since we intend to keep the IEEE 802.15.4 PHY layer, the PHY header with length of 6 octets will be preserved in its original format. In addition, the original beaconing mechanism and beacon frame structure will remain unchanged in our protocol. Therefore, we have focused on reducing the MAC overhead for data frames. For clarity reasons, the definition of the data frame in the IEEE 802.15.4 MAC is shown in Fig. 3.

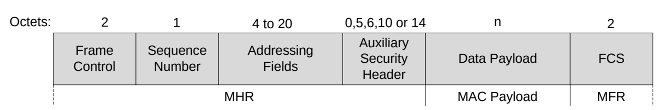

Figure 3. IEEE 802.15.4 MAC data frame format

The MAC header is composed of four fields, among which the optional security field can be removed, because no security aspects are considered in our application scenario. The sequence number is also not needed for unacknowledged GTS transmissions as in our case and can be further ignored. The addressing field specifies the PAN identifier and device address for both, the source and the destination. Because our protocol is designed for managing only one PAN, the PAN identifier field can also be removed. As described previously, all GTSs are preallocated to each device. Thus, the PAN coordinator can easily identify the source of the received message according to the relative position of the GTS in the superframe. Based on this idea, we propose to use an implicit addressing mode instead of the usual explicit scheme utilized in IEEE 802.15.4. Therefore, the addressing field is not necessary for our improved protocol. The frame control filed defines the frame type, addressing mode control flags, and other control flags. The frame type field is not required, for only one type of alarm messages is defined in our application. The addressing mode control flags are useless due to our deployed implicit addressing mode. Other control flags in the frame control filed are unrelated to our application. In a word, the complete frame control field can be removed.

In summary, we propose a new data frame format at the MAC layer that only includes a payload of one byte and a FCS field with 2 bytes in length. The IEEE 802.15.4 MAC header is completely abandoned, resulting an alarm message with only 9 bytes in length including the PHY header and l_D, which is equal to 18 symbols. Compared to the original length of 20 bytes, the overhead in the alarm message has been significantly reduced.

4.3 Performance Analysis

We now reanalyze our improved low-latency protocol version for our studied scenario. According to Fig. 2, (4) is to be used to calculate the new beacon interval. l_{IFS} and l_{SIFS} are set to 12 symbols and 4 symbols, respectively. l_B remains 34 symbols. l_{GTS} is assigned equal to l_D with 18 symbols.

$$l_{BI} = l_B + 2 \times l_{IFS} + n \times l_{GTS} + (n-1) \times l_{SIFS} \tag{4}$$

Based on this, generally d_G can be calculated as follows:

$$l_G = l_B + 2 \times l_{IFS} + n \times l_{GTS} + (n-1) \times l_{SIFS} + l_{TR} \tag{5}$$

According to (4), for 20 devices, l_{BI} equals to 494 symbols. The guaranteed latency achieved by the new protocol is evaluated in the two protocol options:

Beacon tracking enabled – If the device keeps tracking the beacons, no extra latency will be spent on searching for the beacon. The worst case for this option has been discussed in Section 3 and the guaranteed latency can be calculated using (5). l_{TR} is the sum of l_D and l_{SIFS}, and is equal to 22 symbols. Thus, the calculated guaranteed latency for 20 devices is $l_G = 516$ symbols or $d_G = l_G/(62.5 \text{ ksymbols/s}) = 8.3$ ms, which satisfies our requirements.

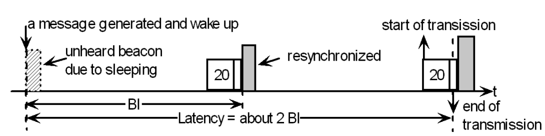

Figure 4. Worst case for beacon tracking disabled

Beacon tracking disabled – The worst case for this option is shown in Fig. 4. The device allocated with the last GTS in the superframe generates a new alarm message and wakes up to listen for a beacon. If this device wakes up right after it has past the first bit of an ongoing beacon transmission, it has to wait an extra beacon interval for the next beacon to arrive. Upon reception of the beacon, the device has to delay the transmission until the arrival of its GTS. In this worst case, the generated message has to wait approximately two beacon intervals before its transmission. Therefore, the guaranteed latency can be estimated as the transmission time for $2 \times l_{BI}$, which equals to 15.81 ms. Although this value exceeds the required 10 ms, it can be deployed in the applications that stress more energy-efficiency than low latency.

5 Conclusion

We evaluated the applicability of the IEEE 802.15.4 protocol for industrial automation scenarios with strict real-time requirements. Using analytical techniques, we identified some restricting limitations of the standard protocol. Based on our findings, we proposed an improved version of the IEEE 802.15.4 MAC layer that keeps the original PHY layer. The improvements include a modified superframe structure supporting only GTS allocations and a new data frame format. Our solution allows the network working in either beacon-tracking enabled or disabled mode, which result in different energy consumption levels. The analysis results have shown that the required guaranteed latency bounds can be satisfied for the 20 devices example when the beacon tracking is enabled. The derived formulas can be used to calculate the maximum number of nodes for a given latency bound as well as to estimate the maximum data latency for a given network size.

References

1. P. Baronti, P. Pillai, V. W. Chook, S. Chessa, A. Gotta, and Y. F. Hu. Wireless Sensor Networks: a Survey on the State of the Art and the 802.15.4 and ZigBee Standards. *Elsevier Computer Communications*, 30(7):1655–1695, May 2007.
2. IEEE. Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks (WPANs). IEEE Sandard 802.15.4-2006, 2006.
3. T. H. Kim and S. Choi. Priority-Based Delay Mitigation for Event-Monitoring IEEE 802.15.4 LR-WPANs. *IEEE Communications Letters*, 10(3):213–215, March 2006.
4. A. Koubaa, M. Alves, and E. Tovar. GTS Allocation Analysis in IEEE 802.15.4 for Real-time Wireless Sensor Networks. In *20th International Parallel and Distributed Processing Symposium (IPDPS 2006)*, April 2006.
5. A. Koubaa, M. Alves, and E. Tovar. i-GAME: An Implicit GTS Allocation Mechanism in IEEE 802.15.4 for Time-Sensitive Wireless Sensor Networks. In *18th Euromicro Conference on Real-Time Systems (ECRTS'06)*, pages 183–192, July 2006.
6. M. Petrova, J. Riihijärvi, P. Mähönen, and S. Labella. Performance Study of IEEE 802.15.4 Using Measurements and Simulations. In *IEEE Wireless Communications and Networking Conference (WCNC 2006)*, pages 487–492, Las Vegas, NV, USA, April 2006.
7. W. Ye, J. Heidemann, and D. Estrin. An Energy-Efficient MAC Protocol for Wireless Sensor Networks. In *21st IEEE Conference on Computer Communications (IEEE INFOCOM 2002)*, volume 3, pages 1567–1576, New York, NY, USA, June 2002.
8. J. Zheng and M. J. Lee. A Comprehensive Performance Study of IEEE 802.15.4. In *Sensor Network Operations*, pages 218–237. IEEE Press, 2006.